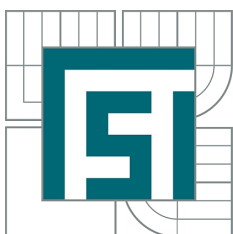


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ  
ÚSTAV MATEMATIKY  
FACULTY OF MECHANICAL ENGINEERING  
INSTITUTE OF MATHEMATICS

# NUMERICKÉ METODY REGISTRACE OBRAZŮ S VYUŽITÍM NELINEÁRNÍ GEOMETRICKÉ TRANSFORMACE

NUMERICAL METHOD OF IMAGE REGISTRATION USING NONLINEAR GEOMETRIC  
TRANSFORM

DIPLOMOVÁ PRÁCE  
MASTER'S THESIS

AUTOR PRÁCE  
AUTHOR

Bc. JAKUB STODOLA

VEDOUCÍ PRÁCE  
SUPERVISOR

doc. Ing. PAVEL ŠTARHA, Ph.D.



Vysoké učení technické v Brně, Fakulta strojního inženýrství

Ústav matematiky

Akademický rok: 2014/2015

## **ZADÁNÍ DIPLOMOVÉ PRÁCE**

student(ka): Bc. Jakub Stodola

který/která studuje v **magisterském navazujícím studijním programu**

obor: **Matematické inženýrství (3901T021)**

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

**Numerické metody registrace obrazů s využitím nelineární geometrické transformace**

v anglickém jazyce:

**Numerical Method of Image Registration Using Nonlinear Geometric Transform**

Stručná charakteristika problematiky úkolu:

Při pozorování objektů přes atmosféru Země dochází vlivem její nehomogenity ke geometrické deformaci obrazu. Tato nehomogenita je v čase proměnná, tedy je proměnná i deformace. Pro registraci (sesazení) více snímků je potřeba tuto deformaci korigovat.

Cíle diplomové práce:

Vytvoření aplikace, která využije metody registrace obrazů s použitím nelineárních geometrických transformací jednotlivých obrazů.

Seznam odborné literatury:

M. Klíma, M. Bernas, J. Hozman, P. Dvořák: Zpracování obrazové informace. CVUT Praha.  
William, K. Pratt.: Digital image processing (2nd ed.). John Wiley & Sons, Inc. New York, NY, USA 1991.

Vedoucí diplomové práce: Ing. Pavel Štarha, Ph.D.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2014/2015.

V Brně, dne 14.11.2014

L.S.

---

prof. RNDr. Josef Šlapal, CSc.  
Ředitel ústavu

---

doc. Ing. Jaroslav Katolický, Ph.D.  
Děkan fakulty

## **Abstrakt**

Cílem práce je najít obecnou nelineární geometrickou transformaci, která kompenzuje nepravidelnou deformaci obrazů, tak aby obrazy bylo možné sesadit. V úvodní části je uveden potřebný matematický aparát, zejména operátory konvoluce, korelace a Fourierovy transformace. V další části je uvedena metoda fázové korelace. Následně jsou uvedeny algoritmy, kterými se geometrická transformace najde. Tyto algoritmy jsou implementovány v počítačovém programu, který je součástí práce.

## **Summary**

The goal of the thesis is to find general nonlinear geometric transformation, which compensates irregular deformation of images, so that they could be registered. In the introductory part, necessary mathematical tools are stated, especially convolution, correlation and Fourier transform. In the next part, method of phase correlation is stated, followed by algorithms used for finding the geometric transformation. Those algorithms are implemented in computer program, that is included.

## **Klíčová slova**

registrace obrazů, fázová korelace, Fourierova transformace, geometrická transformace, konvoluce, korelace, analýza obrazu

## **Keywords**

image registration, phase correlation, Fourier transform, geometric transformation, convolution, correlation, image analysis

STODOLA, J. *Numerické metody registrace obrazů s využitím nelineární geometrické transformace*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2015. 32 s. Vedoucí doc. Ing. Pavel Štarha, Ph.D.



Prohlašuji, že jsem diplomovou práci na téma Numerické metody registrace obrazů s využitím nelineární geometrické transformace vypracoval samostatně s použitím literatury a pramenů, které jsou uvedeny v seznamu na konci této práce.

Bc. Jakub Stodola





Děkuji vedoucímu mé diplomové práce doc. Ing. Pavlu Štarhovi, Ph.D. hlavně za to, že ochotně akceptoval moji žádost o změnu zadání, když jsem se pro toto téma rozhodl až po termínu volby. V té době jsem ještě neměl o této problematice přehled. Dále bych chtěl poděkovat prof. RNDr. Miloslavu Druckmüllerovi, CSc. za znalosti získané z jeho naplňujících přednášek obrazové analýzy, které jsem při psaní této práce využil.

Bc. Jakub Stodola



# Obsah

<b>1 Úvod</b>	<b>3</b>
<b>2 Digitální obraz</b>	<b>4</b>
2.1 Obrazová matice . . . . .	4
<b>3 Operátory konjugace, reflexe a involuce</b>	<b>5</b>
3.1 Konjugace . . . . .	5
3.2 Reflexe . . . . .	5
3.3 Involuce . . . . .	5
<b>4 Kovoluce a korelace</b>	<b>6</b>
4.1 Konvoluce . . . . .	6
4.2 Korelace . . . . .	6
4.3 Diskrétní konvoluce a korelace . . . . .	7
<b>5 Fourierova transformace</b>	<b>8</b>
5.1 Definice - spojitý případ . . . . .	8
5.2 Dvojměrná Fourierova transformace - spojitý případ . . . . .	8
5.3 Diskrétní Fourierova transformace . . . . .	9
5.4 Dvojměrná diskrétní Fourierova transformace . . . . .	9
5.5 Rychlá Fourierova transformace . . . . .	9
<b>6 Věty o Fourierově Transformaci, konvoluci a korelaci</b>	<b>11</b>
6.1 Aditivita . . . . .	11
6.2 O posuvu . . . . .	11
6.3 Konvoluční teorém . . . . .	12
6.4 Věta o korelaci . . . . .	13
<b>7 Fázová korelace</b>	<b>14</b>
<b>8 Geometrické transformace</b>	<b>16</b>
<b>9 Použité algoritmy</b>	<b>17</b>
9.1 Hledání lokálních posuvů pomocí fázové korelace . . . . .	17
9.2 Konvoluce s gaussovským jádrem . . . . .	17
9.3 Sesazení většího počtu obrazů . . . . .	18
9.4 Poznámka o úplné korelaci . . . . .	21
9.5 Rozptyl hodnot a histogram . . . . .	21
9.6 Okna . . . . .	23
<b>10 Popis programu</b>	<b>24</b>
10.1 Uživatelské rozhraní . . . . .	24
<b>11 Výsledky</b>	<b>26</b>
11.1 Globální sesazení více obrazů na průměr . . . . .	26
11.2 Lokální i globální sesazení zároveň, oba typy korelace . . . . .	27

## *OBSAH*

<b>12 Závěr</b>	<b>29</b>
<b>13 Seznam použitých zkratek a symbolů</b>	<b>31</b>
<b>14 Seznam příloh</b>	<b>32</b>

# 1. Úvod

Cílem práce je vytvoření počítačové aplikace, která sesadí dané obrazy s použitím nelineárních geometrických transformací. Aplikace má fungovat obecně pro různá vstupní data, jako příklad daných obrazů jsem dostal fotografie Měsíce, vyfotografované skrz atmosféru Země.

Úvodní kapitoly shrnují matematický aparát, který bude následně použit při řešení daného úkolu. Máme na mysli především konvoluci, korelaci a Fourierovu transformaci, většina uvedených poznatků vychází z [7], [2] a [3]. Dále je obsažena kapitola o větách, které platí pro konvoluci, korelaci a Fourierovu transformaci, ta vychází hlavně z [8]. Další část je věnována fázové korelaci. Něco málo o dalších využitích fázové korelace můžeme najít v [5] a [6]. Následující část popisuje algoritmy, použité pro nalezení geometrické transformace. Tyto algoritmy jsou implementovány v počítačovém programu, který je velmi důležitou součástí práce. V závěrečných kapitolách je uživatelský popis programu a dosažené výsledky.

## 2. Digitální obraz

### 2.1. Obrazová matice

Dvojměrnou spojitou obrazovou informaci si můžeme představit jako intenzitu jasu rozloženou na ploše. Abychom mohli obrazovou informaci zpracovávat počítačem, musíme ji převést na digitální (tj. číslicovou) podobu, které dále budeme říkat *obrazová matice*. Převod spojitě obrazové informace na obrazovou matici nazveme *digitalizací*.

**Definice 2.1.** Funkci  $f : (a, b) \times (c, d) \rightarrow \mathbb{C}$ , kde  $(a, b), (c, d) \subset \mathbb{R}$ , nazveme *spojitou dvojměrnou obrazovou funkcí* nebo zkráceně *obrazovou funkcí*.

**Poznámka.** Pro snadnou představu a použití jsme zadefinovali funkci  $f(x, y)$  na obdélníku, aby byla snadno převeditelná na obrazovou matici. Jako obor hodnot definujeme komplexní čísla. Reálná část komplexní hodnoty představuje intenzitu jasu. Imaginární část budeme potřebovat při Fourierově transformaci.

Obrazovou maticí myslíme matici o  $M \times N$  prvcích. Za jednotlivé prvky bereme komplexní čísla. Jedná se tedy o diskrétní podobu obrazové funkce. Převodem obrazové informace na obrazovou matici se nyní nebudeme zabývat, protože budeme zpracovávat data (fotografie), která už jsou digitalizovaná (ve formě bitmapy).

Obrazovou funkci jsme definovali kvůli lepší představě obrazové matice. K teoretickým odvozování konvoluce, Fourierovy transformace a a fázové korelace, která budou platit pro spojitý případ, budeme používat funkce definované na celém  $\mathbb{R}^2$ . Tyto postupy se následně pokusíme diskretizovat, aby byly použitelné pro zpracování digitalizovaných dat.

## 3. Operátory konjugace, reflexe a involuce

### 3.1. Konjugace

**Definice 3.1.** Funkci  $\overline{f(t)}$  nazveme funkcí *konjugovanou* k funkci  $f(t)$ , jestliže její funkční hodnoty jsou čísla komplexně sdružená (tj. mají opačnou imaginární část) k odpovídajícím funkčním hodnotám funkce  $f(t)$ .

### 3.2. Reflexe

**Definice 3.2.** Operátor reflexe  $Rf$ .

$$Rf(t) = f(-t) \quad (3.1)$$

### 3.3. Involuce

**Definice 3.3.** Operátor involuce  $\widetilde{f(t)}$ .

$$\widetilde{f(t)} = \overline{f(-t)} = \overline{Rf(t)} \quad (3.2)$$

Fourierova transformace je uvedena až v kapitole 5. Pro přehlednost jsou vlastnosti Fourierovy transformace funkcí s těmito operátory uvedeny už zde.

**Věta 3.4.** Platí následující vztahy:

$$\text{Fourierova transformace z reflexe} \quad \mathcal{F}\{Rf\} = R\mathcal{F}\{f\}$$

$$\text{Fourierova transformace z konjugace} \quad \mathcal{F}\{\overline{f}\} = \widetilde{\mathcal{F}\{f\}}$$

$$\text{Fourierova transformace z involuce} \quad \mathcal{F}\{\widetilde{f}\} = \overline{\mathcal{F}\{f\}}$$

Tyto vlastnosti se dají dokázat triviálně přímým dosazením do vzorce Fourierovy transformace, proto důkazy není třeba uvádět.

## 4. Kovoluce a korelace

V definicích konvoluce a korelace uvažujeme absolutně integrovatelné funkce z prostoru

$$\left\{ f|f : \mathbb{R} \rightarrow \mathbb{C}, \int_{\mathbb{R}} |f(t)|dt < \infty \right\} \quad (4.1)$$

Integrál v definici musí existovat a musí být konečný pro skoro všechna  $t \in \mathbb{R}$ .

### 4.1. Konvoluce

**Definice 4.1.** Konvoluce dvou funkcí  $f(t)$  a  $g(t)$  je definována jako

$$f(t) * g(t) = \int_{-\infty}^{\infty} f(\tau)g(t - \tau)d\tau \quad (4.2)$$

V případě funkcí dvou proměnných konvoluci definujeme obdobně.

$$f(x, y) * g(x, y) = \int_{\mathbb{R}^2} f(u, v)g(x - u, y - v)dudv \quad (4.3)$$

Vlastnosti konvoluce:

komutativita	$f * g = g * f$
násobení konstantou	$C_1 f * C_2 g = C_1 C_2 (f * g)$
distributivita	$f * (g_1 + g_2) = f * g_1 + f * g_2$
asociativita	$f * (g * h) = (f * g) * h$

### 4.2. Korelace

**Definice 4.2.** Korelace dvou funkcí  $f(t)$  a  $g(t)$  je definována jako

$$f(t) \times g(t) = \int_{-\infty}^{\infty} \overline{f(\tau)}g(t + \tau)d\tau \quad (4.4)$$

V případě funkcí dvou proměnných, korelaci definujeme obdobně.

$$f(x, y) \times g(x, y) = \int_{\mathbb{R}^2} \overline{f(u, v)}g(x + u, y + v)dudv \quad (4.5)$$

Vlastnosti korelace:

komutativita	$f \times g = \widetilde{g \times f}$
násobení konstantou	$Cf \times g = \overline{C}(f \times g)$ $f \times Cg = C(f \times g)$
distributivita	$f \times (g_1 + g_2) = f \times g_1 + f \times g_2$ $(f_1 + f_2) \times g = f_1 \times g + f_2 \times g$
asociativita	$f \times (g \times h) = (\widetilde{f \times g}) \times h$

Vztah mezi konvolucí a korelací:

$$f \times g = \widetilde{f} * g \quad (4.6)$$



### 4.3. Diskrétní konvoluce a korelace

Nechť v následujících definicích jsou  $f = \{f_m\}_{m=-\infty}^{\infty}$ ,  $g = \{g_m\}_{m=-\infty}^{\infty}$ ,  $h = \{h_m\}_{m=-\infty}^{\infty}$  posloupnosti z prostoru  $\{\xi \mid \xi = \{\xi_m\}_{m \in \mathbb{Z}}, \sum_{m \in \mathbb{Z}} |\xi_m| < \infty, \xi_m \in \mathbb{C}\}$ . Tj. prostor konečně sumovatelných posloupností z  $\mathbb{C}^{\mathbb{Z}}$ . Více o prostorech posloupností se můžeme dozvědět v textech zabývajících se funkcionální analýzou, např. v [1] a [7]. Více o odvození diskrétní podoby konvoluce a korelace nalezneme v [7].

**Definice 4.3.** Existují-li a jsou-li konečné následující sumy

$$h_i = \sum_{m=-\infty}^{\infty} f_m g_{i-m}, \quad (4.7)$$

jejich posloupnost  $h = \{h_i\}_{i=-\infty}^{\infty}$  nazveme *diskrétní konvolucí posloupností  $f, g$*  a píšeme  $h = (f * g)$ .

**Definice 4.4.** Za stejných předpokladů jako v předchozí definici definujeme *diskrétní korelaci posloupností*

$$h_i = \sum_{m=-\infty}^{\infty} \overline{f_m} g_{i+m} \quad (4.8)$$

a píšeme  $h = (f \times g)$

**Definice 4.5.** Jsou-li posloupnosti  $f, g$   $N$ -periodické s opakujícími se částmi určenými vektory  $\mathbf{f} = (f_0, f_1, \dots, f_{N-1})$ ,  $\mathbf{g} = (g_0, g_1, \dots, g_{N-1})$ , pak *diskrétní cyklická konvoluce* je  $N$ -periodická posloupnost  $h$  s opakující se částí  $\mathbf{h} = (h_0, h_1, \dots, h_{N-1})$ , kde

$$h_i = \frac{1}{N} \sum_{m=0}^{N-1} f_m g_{i-m}. \quad (4.9)$$

Budeme ji značit stejně jako necyklickou  $h = f * g$ .

V případě dvojrozměrné obrazové matice tento vztah upravíme na

$$h_{i,j} = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f_{m,n} g_{i-m,j-n} \quad (4.10)$$

**Definice 4.6.** Za stejných předpokladů jako v (4.9) definujeme *diskrétní cyklickou korelaci*.

$$h_i = \frac{1}{N} \sum_{m=0}^{N-1} \overline{f_m} g_{i+m} \quad (4.11)$$

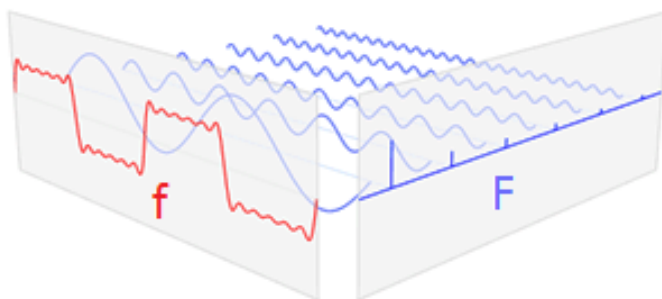
Značení opět ponecháme stejné  $h = f \times g$ .

V případě dvojrozměrné obrazové matice tento vztah upravíme na

$$(f \times g)_{i,j} = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \overline{f_{m,n}} g_{i+m,j+n} \quad (4.12)$$

## 5. Fourierova transformace

Fourierova transformace je integrální transformace, která kromě obrazové analýzy slouží například pro práci se signály. Její funkční hodnoty jsou komplexní čísla, jejichž absolutní hodnota může být chápána jako amplituda základní sinusoidy určité frekvence a komplexní argument jako její fázový posun. Původní funkce je tedy součtem (teoreticky nekonečně mnoha) těchto základních sinusoid. Znázornění je na následujícím obrázku.



Obrázek 5.1: Znázornění Fourierovy transformace

Na obrázku vidíme Fourierovu transformaci, která funkci  $f$  přiřazuje funkci  $F$ , značíme  $\mathcal{F}\{f\} = F$ . Zpětnou Fourierovu transformaci značíme  $\mathcal{F}^{-1}\{F\} = f$ .

**Poznámka.** V této kapitole uvažujeme funkce a posloupnosti ze stejných prostorů jako v předchozí kapitole.

### 5.1. Definice - spojitý případ

**Definice 5.1.** Fourierova transformace (dopředná)

$$F(\xi) = \int_{-\infty}^{\infty} f(t)e^{-2\pi i \xi t} dt \quad (5.1)$$

**Definice 5.2.** Inverzní (zpětná) Fourierova transformace

$$f(t) = \int_{-\infty}^{\infty} F(\xi)e^{2\pi i \xi t} d\xi \quad (5.2)$$

### 5.2. Dvojměrná Fourierova transformace - spojitý případ

**Definice 5.3.** Fourierova transformace (dopředná)

$$F(u, v) = \int_{\mathbb{R}^2} f(x, y)e^{-2\pi i(xu+yv)} dx dy \quad (5.3)$$

**Definice 5.4.** Inverzní (zpětná) Fourierova transformace

$$f(x, y) = \int_{\mathbb{R}^2} F(u, v) e^{2\pi i(xu+yv)} du dv \quad (5.4)$$

### 5.3. Diskrétní Fourierova transformace

**Definice 5.5.** Diskrétní Fourierova transformace (dopředná)

$$F_k = \sum_{n=0}^{N-1} f_n e^{\frac{-2\pi i k n}{N}} \quad (5.5)$$

**Definice 5.6.** Inverzní (zpětná) diskrétní Fourierova transformace

$$f_n = \frac{1}{N} \sum_{k=0}^{N-1} F_k e^{\frac{2\pi i k n}{N}} \quad (5.6)$$

### 5.4. Dvojměrná diskrétní Fourierova transformace

**Definice 5.7.** Dvojměrná diskrétní Fourierova transformace (dopředná)

$$F_{u,v} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f_{m,n} e^{-2\pi i(\frac{mu}{M} + \frac{nv}{N})} \quad (5.7)$$

**Definice 5.8.** Inverzní (zpětná) dvojměrná diskrétní Fourierova transformace

$$f_{m,n} = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F_{u,v} e^{2\pi i(\frac{mu}{M} + \frac{nv}{N})} \quad (5.8)$$

### 5.5. Rychlá Fourierova transformace

Při implementaci Fourierovy transformace musíme uvažovat výpočetní náročnost algoritmu (časovou nebo paměťovou). Operační náročností se myslí počet operací, který algoritmus vykoná v závislosti na počtu vstupních dat  $N$ . Asymptotická složitost  $\mathcal{O}(f(N))$  popisuje zjednodušeně operační náročnost algoritmu pomocí „nejvyššího řádu“ funkce  $f(N)$ .

**Definice 5.9.** Řekneme, že algoritmus má asymptotickou složitost  $\mathcal{O}(f(N))$ , jestliže  $\exists C \in (0, \infty)$ ,  $N_0 \in \mathbb{N}$  taková, že platí  $g(N) \leq C f(N) \forall N > N_0$ , kde  $g(N)$  je počet operací, který algoritmus vykoná v závislosti na počtu vstupních dat  $N$ .

Jednorozměrná diskrétní Fourierova transformace má kvadratickou asymptotickou složitost  $\mathcal{O}(N^2)$ . Při výpočtu jednoho bodu  $F_k$ , viz 5.5 probíhá sumace přes  $N$  bodů, tedy výpočet jednoho bodu  $F_k$  má asymptotickou složitost  $\mathcal{O}(N)$ .  $F_k$  musíme počítat  $N$ -krát, protože  $k = 0, 1, \dots, N-1$ . Celkem proto máme  $N \cdot N$ , tedy  $\mathcal{O}(N^2)$ .

## 5.5. RYCHLÁ FOURIEROVA TRANSFORMACE

**Příklad 5.10.** V reálném případě můžeme mít běžně obraz s rozlišením například  $1000 \times 1000$  pixelů. Když teď opomeneme dva rozměry, zaměříme se pouze na počet dat, potřebujeme zpracovat  $N = 10^6$  dat. V případě kvadratické složitosti Fourierovy transformace potřebujeme vykonat přibližně  $10^{12}$  operací. Kdyby jedna operace trvala  $1\text{ ns}$ , potom by výpočet trval přibližně 17 minut.

Protože to je pomalé, použijeme algoritmus rychlé Fourierovy transformace, zkráceně FFT z anglického fast Fourier transform. FFT má asymptotickou složitost  $\mathcal{O}(N \log_2 N)$ . Výpočet příkladu 5.10 by pomocí FFT trval přibližně 0,02 s.

**Princip FFT** Ukážeme si jeden z možných postupů FFT. Vycházíme ze vzorce 5.5. Posloupnost vstupů  $f_n$  rozložíme na dvě posloupnosti  $f^e$  a  $f^o$ . Posloupnost  $f^e$  je posloupnost  $f_n$  vybraná na sudých  $n$ ,  $f^o$  na lichých  $n$ .

$$\begin{aligned}
 F_k &= \sum_{n=0}^{N-1} f_n e^{\frac{-2\pi i k n}{N}} \\
 &= \sum_{n=0}^{\frac{N}{2}-1} f_{2n} e^{\frac{-2\pi i k (2n)}{N}} + \sum_{n=0}^{\frac{N}{2}-1} f_{2n+1} e^{\frac{-2\pi i k (2n+1)}{N}} \\
 &= \sum_{n=0}^{\frac{N}{2}-1} f_{2n} e^{\frac{-2\pi i k n}{\frac{N}{2}}} + e^{\frac{-2\pi i k}{N}} \sum_{n=0}^{\frac{N}{2}-1} f_{2n+1} e^{\frac{-2\pi i k n}{\frac{N}{2}}} \\
 &= F_k^e + e^{\frac{-2\pi i k}{N}} F_k^o
 \end{aligned} \tag{5.9}$$

$F^e$  a  $F^o$  jsou obrazy  $f^e$  a  $f^o$ . Vidíme tedy, že ze dvou transformací o  $\frac{N}{2}$  prvcích takto rychle získáme hodnotu transformace o  $N$  prvcích. Pokud máme  $N = 2^a$ ,  $a \in \mathbb{N}$ , pak můžeme postupovat rekurzivně a posloupnost  $a$ -krát takto rozdělit. (Počet kroků pro výpočet jednoho  $F_k$  se tedy snížil na  $a = \log_2 N$ .) Tím se dostaneme na jednoprvkovou posloupnost. Zbývá tedy zodpovědět otázku, jaká je hodnota Fourierovy transformace z jednoprvkové posloupnosti.

**Tvrzení 5.11.** Diskrétní Fourierova transformace z jednoprvkové posloupnosti  $f_n = \{f_0\}$  je identita.

*Důkaz.* Dáno  $N = 1$ ,  $f_n = \{f_0\}$ . Přímým dosazením dostaneme

$$F_k = \sum_{n=0}^{N-1} f_n e^{\frac{-2\pi i k n}{N}} = f_0 e^0 = f_0 \tag{5.10}$$

□

Algoritmus můžeme ještě usnadnit uchováním a užitím částečných výpočtů například pomocí tzv. motýlkového schématu. Do podrobností nebudeme dále zacházet.

## 6. Věty o Fourierově Transformaci, konvoluci a korelaci

**Poznámka.** Věty uvedené v této kapitole platí nejen pro funkce, ale i pro posloupnosti. Důkazy v prostorech posloupností by byly analogické k důkazům ve spojitém případě. Proto je pro stručnost nebudeme uvádět.

### 6.1. Aditivita

**Věta 6.1.**

$$\mathcal{F}\{f + g\} = F + G \quad (6.1)$$

*Důkaz.*

$$\begin{aligned} \mathcal{F}\{f + g\}(\xi) &= \int_{-\infty}^{\infty} [f(t) + g(t)] e^{-2\pi i \xi t} dt = \\ &= \int_{-\infty}^{\infty} f(t) e^{-2\pi i \xi t} dt + \int_{-\infty}^{\infty} g(t) e^{-2\pi i \xi t} dt \end{aligned}$$

□

### 6.2. O posuvu

**Věta 6.2.**

$$\mathcal{F}\{f(t - t_0)\}(\xi) = e^{-2\pi i \xi t_0} F(\xi) \quad (6.2)$$

*Důkaz.*

$$\mathcal{F}\{f(t - t_0)\}(\xi) = \int_{-\infty}^{\infty} f(t - t_0) e^{-2\pi i \xi t} dt$$

Integrand vynásobíme  $e^{-2\pi i \xi t_0} e^{2\pi i \xi t_0} = 1$ .

$$\begin{aligned} &= \int_{-\infty}^{\infty} f(t - t_0) e^{-2\pi i \xi t} e^{-2\pi i \xi t_0} e^{2\pi i \xi t_0} dt \\ &= e^{-2\pi i \xi t_0} \int_{-\infty}^{\infty} f(t - t_0) e^{-2\pi i \xi (t - t_0)} dt \end{aligned}$$

Použijeme substituci  $s = t - t_0$ ,  $ds = dt$ .

$$= e^{-2\pi i \xi t_0} \int_{-\infty}^{\infty} f(s) e^{-2\pi i \xi s} ds = e^{-2\pi i \xi t_0} F(\xi)$$

□

**Věta 6.3.**

$$\mathcal{F}^{-1}\{F(\xi - \xi_0)\}(t) = e^{2\pi i \xi_0 t} f(t) \quad (6.3)$$

### 6.3. KONVOLUČNÍ TEORÉM

*Důkaz.*

$$\mathcal{F}^{-1}\{F(\xi - \xi_0)\}(t) = \int_{-\infty}^{\infty} F(\xi - \xi_0) e^{2\pi i \xi t} dt$$

Integrand vynásobíme  $e^{-2\pi i \xi_0 t} e^{2\pi i \xi_0 t} = 1$ .

$$\begin{aligned} &= \int_{-\infty}^{\infty} F(\xi - \xi_0) e^{2\pi i \xi t} e^{-2\pi i \xi_0 t} e^{2\pi i \xi_0 t} dt \\ &= e^{2\pi i \xi_0 t} \int_{-\infty}^{\infty} F(\xi - \xi_0) e^{2\pi i (\xi - \xi_0) t} dt \end{aligned}$$

Použijeme substituci  $s = \xi - \xi_0$ ,  $ds = d\xi$ .

$$= e^{2\pi i \xi_0 t} \int_{-\infty}^{\infty} F(s) e^{2\pi i s t} ds = e^{2\pi i \xi_0 t} f(t)$$

□

Věty o posuvu 6.2, 6.3 platí i pro funkce více proměnných.

$$\mathcal{F}\{f(x - \Delta x, y - \Delta y)\}(u, v) = e^{-2\pi i (u\Delta x + v\Delta y)} F(u, v) \quad (6.4)$$

$$\mathcal{F}^{-1}\{F(u - \Delta u, v - \Delta v)\}(x, y) = e^{2\pi i (\Delta u x + \Delta v y)} f(x, y) \quad (6.5)$$

Důkaz je analogický.

## 6.3. Konvoluční teorém

**Věta 6.4.**

$$\mathcal{F}\{f * g\} = F \cdot G \quad (6.6)$$

*Důkaz.*

$$\begin{aligned} \mathcal{F}\{f * g\}(\xi) &= \int_{-\infty}^{\infty} \left[ \int_{-\infty}^{\infty} f(\tau) g(t - \tau) d\tau \right] e^{-2\pi i \xi t} dt \\ &= \int_{-\infty}^{\infty} f(\tau) \left[ \int_{-\infty}^{\infty} g(t - \tau) e^{-2\pi i \xi t} dt \right] d\tau \end{aligned}$$

Pomocí věty o posuvu 6.2 máme

$$\left[ \int_{-\infty}^{\infty} g(t - \tau) e^{-2\pi i \xi t} dt \right] = e^{-2\pi i \xi \tau} G(\xi)$$

Takže máme

$$\begin{aligned} \mathcal{F}\{f * g\}(\xi) &= \int_{-\infty}^{\infty} f(\tau) e^{-2\pi i \xi \tau} G(\xi) d\tau \\ &= G(\xi) \int_{-\infty}^{\infty} f(\tau) e^{-2\pi i \xi \tau} d\tau = G(\xi) \cdot F(\xi) \end{aligned}$$

□

## 6. VĚTY O FOURIEROVĚ TRANSFORMACI, KONVOLUCI A KORELACI

**Věta 6.5.**

$$\mathcal{F}^{-1}\{F * G\} = f \cdot g \quad (6.7)$$

*Důkaz.*

$$\begin{aligned} \mathcal{F}^{-1}\{F * G\}(t) &= \int_{-\infty}^{\infty} \left[ \int_{-\infty}^{\infty} F(\tau) G(\xi - \tau) d\tau \right] e^{2\pi i \xi t} d\xi \\ &= \int_{-\infty}^{\infty} F(\tau) \left[ \int_{-\infty}^{\infty} G(\xi - \tau) e^{2\pi i \xi t} d\xi \right] d\tau \end{aligned}$$

Pomocí věty o posuvu 6.3 máme

$$\left[ \int_{-\infty}^{\infty} G(\xi - \tau) e^{2\pi i \xi t} d\xi \right] = e^{2\pi i t \tau} g(t)$$

Takže máme

$$\begin{aligned} \mathcal{F}^{-1}\{F * G\}(t) &= \int_{-\infty}^{\infty} F(\tau) e^{2\pi i t \tau} g(t) d\tau \\ &= g(t) \int_{-\infty}^{\infty} F(\tau) e^{2\pi i t \tau} d\tau = g(t) \cdot f(t) \end{aligned}$$

□

### 6.4. Věta o korelaci

**Věta 6.6.**

$$\mathcal{F}\{f \times g\} = \overline{F} \cdot G \quad (6.8)$$

*Důkaz.* Použijeme vztah mezi konvolucí a korelací (4.6) a konvoluční teorém (6.6).

$$\begin{aligned} \mathcal{F}\{f \times g\} &= \mathcal{F}\{\tilde{f} * g\} = \mathcal{F}\{\tilde{f}\} \cdot \mathcal{F}\{g\} = \\ &= \mathcal{F}\{\overline{Rf}\} \cdot G = \mathcal{F}\{R\tilde{f}\} \cdot G = R\mathcal{F}\{\tilde{f}\} \cdot G = \\ &= \widetilde{R\mathcal{F}\{f\}} \cdot G = R\tilde{F} \cdot G = RR\overline{F} \cdot G = \overline{F} \cdot G \end{aligned}$$

□

## 7. Fázová korelace

Pokud jsou dvě funkce stejné až na posuv ( $f(t) = g(t + \Delta t)$ ) můžeme tento posuv zřejmě snadno najít, jako  $\operatorname{argmax}(f \times g)$ . Tento výpočet je ale v diskretizovaném případě výpočetně náročný. Proto bývá lepší použít fázovou korelaci. Ta je uvedena v této kapitole.

**Definice 7.1.** Funkci

$$R(u, v) = \frac{A(u, v) \overline{B(u, v)}}{|A(u, v) \overline{B(u, v)}|}, \quad (7.1)$$

kde  $\overline{B(u, v)}$  je funkce, jejíž funkční hodnoty jsou komplexně sdružené s funkčními hodnotami funkce  $B(u, v)$ , nazveme *normovaným křížovým spektrem* obrazových funkcí  $a(x, y)$ ,  $b(x, y)$ , když platí  $A(u, v) = \mathcal{F}\{a(x, y)\}$ ,  $B(u, v) = \mathcal{F}\{b(x, y)\}$ .

Vzhledem k tomu, že obrazové matice představují diskretizaci dvojrozměrných obrazových funkcí, budeme součin obrazových matic brát jako násobení po složkách (Hadamardův součin).

**Definice 7.2.** *Hadamardovým součinem* myslíme násobení matic o stejných rozměrech po složkách.

$$c = ab \Leftrightarrow c_{i,j} = a_{i,j} b_{i,j} \quad (7.2)$$

pro všechny složky.

**Definice 7.3.** Mějme obrazové matice  $a, b$  a jejich obrazy  $A = \mathcal{F}\{a\}$ ,  $B = \mathcal{F}\{b\}$  získané pomocí dvojrozměrné diskrétní Fourierovy transformace. Obrazovou matici

$$R = \frac{AB}{|AB|}, \quad (7.3)$$

kde  $AB$  je Hadamardův součin matic a  $\overline{B}$  je matice tvořená prvky komplexně sdruženými s prvky matice  $B$ , nazveme *normovaným křížovým spektrem obrazových matic*  $a, b$ . Výrazem  $|\cdot|$  myslíme absolutní hodnoty složek a podílem myslíme dělení po složkách.

**Definice 7.4.** *Normovanou křížovou korelací* nebo také *fázovou korelací* funkcí  $a(x, y)$ ,  $b(x, y)$  myslíme hledání posuvu

$$(\Delta x, \Delta y) = \operatorname{argmax}_{(x,y)} (\mathcal{F}^{-1}\{R\}), \quad (7.4)$$

kde  $R$  je jejich normované křížové spektrum.

**Myšlenka odvození** Máme dvě funkce, lišící se jen posuvem, tedy

$$a(x - \Delta x, y - \Delta y) = b(x, y) \quad (7.5)$$

Z věty o posuvu pro dvě proměnné (6.4) máme:

$$A(u, v) = \mathcal{F}\{a(x - \Delta x, y - \Delta y)\} = e^{-2\pi i(\Delta x u + \Delta y v)} B(u, v) \quad (7.6)$$

Ve výrazu  $A(u, v) \overline{B(u, v)}$ , který se vyskytuje v normovaném křížovém spektru (7.1), za  $A$  dosadíme (7.6).

$$A(u, v) \overline{B(u, v)} = e^{-2\pi i(\Delta x u + \Delta y v)} B(u, v) \overline{B(u, v)} \quad (7.7)$$



Eulerova formule  $e^{i\varphi}$  představuje jednotkovou kružnici v Gaussově rovině, proto její absolutní hodnota je vždy rovna jedné. Tedy substitucí  $\varphi = -2\pi(\Delta x u + \Delta y v)$  dostaneme  $|e^{-2\pi i(\Delta x u + \Delta y v)}| = 1$ . Vložením obou stran do absolutní hodnoty dostaneme

$$|A(u, v)\overline{B(u, v)}| = |B(u, v)\overline{B(u, v)}|. \quad (7.8)$$

Pro funkční hodnoty funkce  $B(u, v)\overline{B(u, v)}$  platí  $B(u, v) = \alpha + i\beta$ , pak

$$B(u, v)\overline{B(u, v)} = (\alpha + i\beta)(\alpha - i\beta) = \alpha^2 + \beta^2, \quad \forall u, v. \quad (7.9)$$

Takže její funkční hodnoty mají nulovou imaginární část a kladnou reálnou část. Proto u  $|B(u, v)\overline{B(u, v)}|$  můžeme vynechat absolutní hodnotu. Potom máme

$$\frac{B(u, v)\overline{B(u, v)}}{|A(u, v)\overline{B(u, v)}|} = 1 \quad (7.10)$$

Dosazením do výrazu  $\mathcal{F}^{-1}\{R\}$ , který se vyskytuje v 7.4 máme

$$\begin{aligned} \mathcal{F}^{-1}\left\{\frac{A(u, v)\overline{B(u, v)}}{|A(u, v)\overline{B(u, v)}|}\right\} &= \mathcal{F}^{-1}\left\{e^{-2\pi i(\Delta x u + \Delta y v)}\frac{B(u, v)\overline{B(u, v)}}{|A(u, v)\overline{B(u, v)}|}\right\} = \\ &= \mathcal{F}^{-1}\left\{e^{-2\pi i(\Delta x u + \Delta y v)}\right\} = \int_{\mathbb{R}^2} e^{-2\pi i(\Delta x u + \Delta y v)} e^{2\pi i(xu + yv)} dudv = \\ &= \int_{\mathbb{R}^2} e^{2\pi i((x - \Delta x)u + (y - \Delta y)v)} dudv \end{aligned} \quad (7.11)$$

Použijeme-li substituci  $s = x - \Delta x$ ,  $r = y - \Delta y$ , dostáváme vzorec pro inverzní Fourierovu transformaci funkce  $F(u, v) \equiv 1$ . Její výsledek je známá Diracova delta funkce  $\delta(r, s)$ . To je funkce, která je všude rovna nule až na počátek a její hodnota v počátku se limitně blíží nekonečnu. Celkem máme

$$\mathcal{F}^{-1}\{R\} = \delta(r, s) = \delta(x - \Delta x, y - \Delta y) \quad (7.12)$$

V ideálním případě bez šumu tedy dostaneme posunutou delta funkci, která je nulová všude kromě bodu  $(x, y) = (\Delta x, \Delta y)$ . V případě, že jsou obrazy zašuměné, nedostaneme přesně nuly, ale hodnoty blízké nule a jeden vrchol. Hledáme tedy argument maxima, tj. souřadnice bodu, kde je vrchol. Čím více se obrazy podobají, tím je výběr maxima zřetelnější.

## 8. Geometrické transformace

Geometrická transformace je operátor, který vyjadřuje geometrickou změnu obrazu většinou způsobenou vlastnostmi optické zobrazovací soustavy. Pomocí geometrické transformace můžeme takovou změnu vytvořit nebo kompenzovat. Například při pozorování Měsíce skrz nehomogenní atmosféru Země dochází k různým v čase proměnným deformacím.

Za nejjednodušší geometrické transformace můžeme považovat lineární transformace posuv, zvětšení a otočení. Transformace bodu obrazové matice vyjadřující posuv se dá zapsat následovně.

$$(x', y')_i = (x_i + \Delta x, y_i + \Delta y) \quad (8.1)$$

**Definice 8.1.** *Geometrickou transformaci* obrazové matice  $a$  chápeme jako operátor, který vytvoří každý prvek nové matice  $a'$  v závislosti na  $a$ , a budeme ji značit  $a' = GT(a)$

Nalezení obecné nelineární transformace  $GT(a)$ , která bude kompenzovat hledanou geometrickou změnu je poněkud problematické, obzvlášť když nemáme přesně definovaný požadovaný výsledek. Ten navíc často závisí na subjektivním pocitu, například jak určit, co je to „ostřejší obraz“ nebo „zatmění Slunce jak ho vidí lidské oko“? V případě kompenzace nehomogenity atmosféry při pozorování Měsíce máme podobný problém, a to „Jak by to vypadalo bez atmosféry Země?“.

## 9. Použité algoritmy

### 9.1. Hledání lokálních posuvů pomocí fázové korelace

Mějme sesazovanou obrazovou matici  $s$  a referenční obrazovou matici  $r$ . Pokusíme se nalézt geometrickou transformaci  $GT(s)$ , která by aproximovala matici  $r$ . Tj.  $GT(s) \approx r$ . Zavedeme množinu *iniciálních uzlových bodů*  $U = \{u_m\}_{m \in M}$ ,  $M \subset \mathbb{N}_0$ , kde  $u_m = (k, l)$  jsou „základní pozice“ uzlových bodů. Lokální fázovou korelací spočítáme posuvy uzlových bodů  $T(u_m)$  v každé sesazované obrazové matici  $s$ . Matice  $s_m$  (resp.  $r_m$ ) je lokálně vybraná část sesazované (resp. referenční) obrazové matice  $s$  (resp.  $r$ ) v okolí uzlu  $u_m$ .

$$T(u_m) = \operatorname{argmax} \left( \mathcal{F}^{-1} \left\{ \frac{\mathcal{F}\{s_m\} \overline{\mathcal{F}\{r_m\}}}{|\mathcal{F}\{s_m\} \mathcal{F}\{r_m\}|} \right\} \right), \quad m \in M \quad (9.1)$$

Výslednou transformaci  $a = GT(s)$  obdržíme pomocí konvoluce s gaussovským jádrem, viz 9.2.  $T(u_m)$  vyjadřuje vektory posuvů v jednotlivých uzlových bodech, je to tedy vektorové pole. Zbývá otázka, jak volit uzlové body. Ty můžeme volit například jako ekvidistantní síť. Z důvodu uvedeného v 9.5 je lepší volit uzlové body na hranách objektů. Pokud by obrazová matice v okolí uzlu popisovala jednobarevnou plochu, její prvky by měly přibližně stejnou hodnotu a sesazení okolí na okolí by nebylo jednoznačné. V programu jsem místo detekce hran zvolil přístup, že body volím ekvidistantně a pro každý z nich posuzuji, zda je nalezené maximum významné. Oba tyto postupy celkový algoritmus značně zpomalují.

### 9.2. Konvoluce s gaussovským jádrem

Nyní známe lokální posuvy v uzlových bodech. Musíme nalézt způsob, jakým z těchto uzlových bodů odhadneme posuvy ve zbývajících bodech výsledné obrazové matice. Takovému způsobu se říká *interpolace*. Existuje velké množství metod interpolace, například lineární interpolace nebo interpolace polynomem. Těmi se v tomto textu nebudeme zabývat, popíšeme pro nás vhodnější metodu konvoluční filtraci typu dolní propust. Výsledný obraz určíme konvolucí  $a = T(u_m) * g$ , kde  $a$  je výsledná obrazová funkce,  $T(u_m)$  je funkce vyjadřující posuvy v uzlových bodech  $u_m$  a  $g$  je konvoluční jádro. Za konvoluční jádro je nejlepší zvolit gaussovskou funkci.

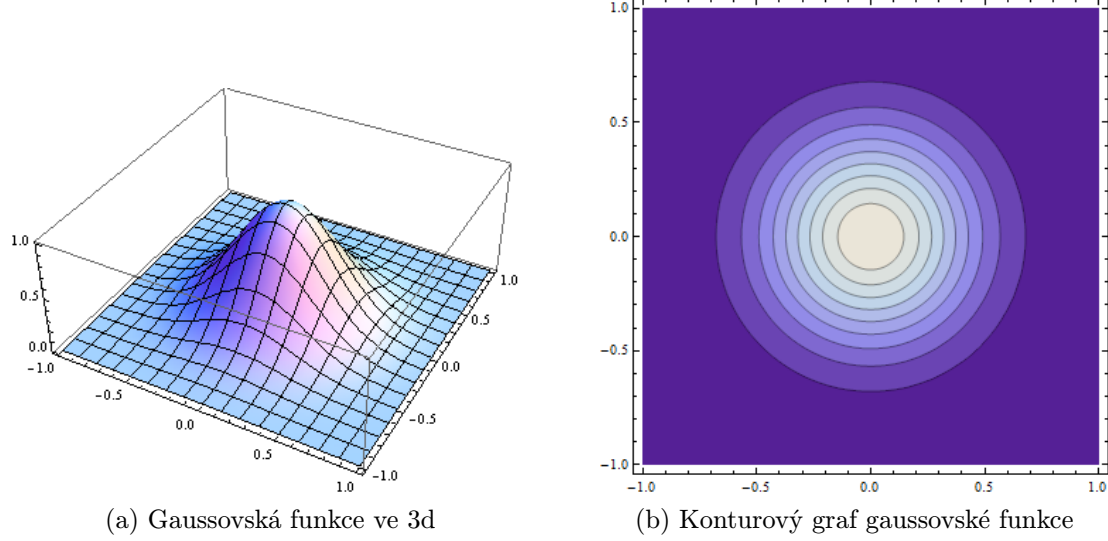
**Definice 9.1.** *Gaussovská funkce* je funkce ve tvaru

$$g(t) = ae^{-\frac{(x-\mu)^2}{2\sigma^2}}. \quad (9.2)$$

Gaussovská funkce hraje důležitou roli ve statistice, kde popisuje normální rozdělení se střední hodnotou  $\mu$  a rozptylem  $\sigma^2$ . V našem případě budeme parametry stanovovat experimentálně. Protože „dobrý“ výsledek závisí na subjektivním dojmu. Nikde není přesně stanoveno, jaký výsledek je nejlepší. V našem případě potřebujeme gaussovskou funkci dvou proměnných, vystačíme si s funkcí

$$g(x, y) = e^{-\frac{x^2+y^2}{s}} \quad (9.3)$$

### 9.3. SESAZENÍ VĚTŠÍHO POČTU OBRAZŮ



Obrázek 9.1: Zobrazení gaussovské funkce

V našem případě diskrétní funkce posuvů v uzlových bodech  $T$  si můžeme tuto konvoluci pro jeden hledaný bod  $a_{i,j}$  představit jako vážený průměr posuvů  $T(u_m)$ , kde váha je závislá na vzdálenosti uzlového bodu  $u_m = (k, l)_m$  od bodu  $a_{i,j}$ . Váha  $w(u_m, a_{i,j}) = g(k - i, l - j)$ , tj. hodnota gaussovské funkce (9.3) se středem v bodě  $a_{i,j}$ . To si můžeme představit z obrázku 9.1b, uzlům dále od středu odpovídá nižší váha zobrazená tmavší barvou. Parametr  $a$  v (9.2) určuje „výšku klobouku“, zvolili jsme ho jednoduše  $a = 1$ , váhy jsou potom  $w(u_m, a_{i,j}) \in \langle 0, 1 \rangle$ . Abychom lépe mohli určit parametr  $s$ , je lepší normovat parametry funkce  $g(x, y)$ , aby  $x, y \in \langle 0, 1 \rangle$ . To zajistíme podělením parametrů rozměry matice. Nechť matice  $a$  má rozměry  $M \times N$ . Pak

$$w(u_m, a_{i,j}) = g\left(\frac{k-i}{M}, \frac{l-j}{N}\right) \quad (9.4)$$

$$a_{i,j} = \frac{\sum_m w(u_m, a_{i,j}) T(u_m)}{\sum w(u_m, a_{i,j})} \quad (9.5)$$

Funkční hodnoty funkce  $T(u_m)$  představují dvojrozměrné vektory posuvů v uzlových bodech  $u_m$ . Funkční hodnoty matice  $a$  jsou dvojrozměrné vektory představující posuvy jednotlivých bodů sesazované obrazové matice  $s$ , aby se podobala referenční matici  $r$ . Platí tedy  $r_{i,j} \approx s_{i+(a_{i,j}^1), j+(a_{i,j}^2)}$ , kde vektor  $(a_{i,j}^1, a_{i,j}^2) = a_{i,j}$ .

### 9.3. Sesazení většího počtu obrazů

Pokud máme více obrazů a chceme je sesadit v jeden, můžeme posuvy uzlových bodů následujícím způsobem zprůměrovat. Jeden z obrazů si nejdříve zvolíme jako referenční. Přičtením posuvů  $T(u_m)_i$  každé sesazované obrazové matice  $s_i$  k souřadnicím iniciálních uzlových bodů  $\{u_m\}$  dostaneme souřadnice *posunutých uzlových bodů*.

$$U_i^P(u_m) = u_m + T_i(u_m) \quad (9.6)$$

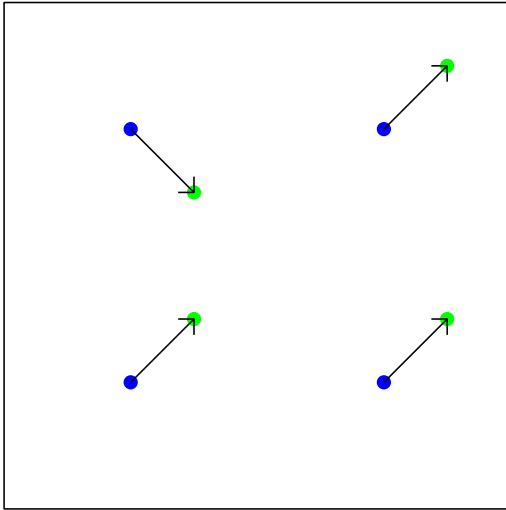
Potom aritmeticky zprůměrujeme souřadnice posunutých uzlových bodů ve všech obrazech včetně referenčního, kde posunuté uzlové body zůstanou na iniciálních hodnotách  $u_m$ .

Tak dostaneme průměr  $\overline{U^P}$  posunutých uzlových bodů  $U_i^P$ . Posuvům v uzlových bodech přiřadíme nové hodnoty

$$T_i := U_i^P - \overline{U^P}, \quad (9.7)$$

které značí posuv iniciálních uzlových bodů v jednotlivých obrazech (včetně referenčního), které obrazy převedou do nového průměrného obrazu. Na obrázku 9.2c máme referenční obraz černě, sesazované obrazy zeleně a oranžově. Vidíme, že původní (barevně znázorněné) vektory  $T_i$  posunují obraz na referenční. Dále vidíme, že nově vypočítané (černě znázorněné) vektory  $T_i$  posunou všechny tři původní obrazy do jednoho místa.

### 9.3. SESAZENÍ VĚTŠÍHO POČTU OBRAZŮ

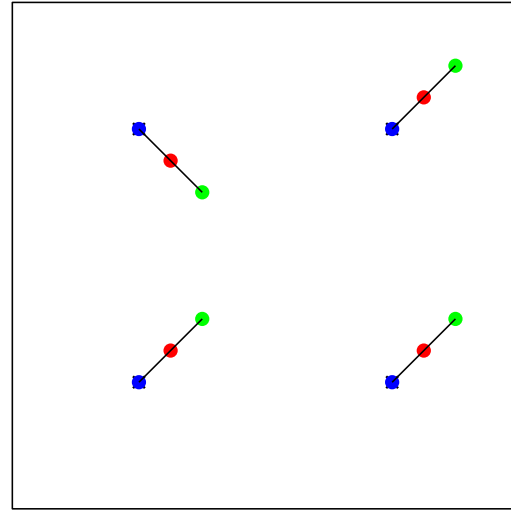


•  $\{u_m\}$

•  $U_1^P$

$\nearrow T_1$

(a) Iničiální uzlové body, posuvy a posunuté body

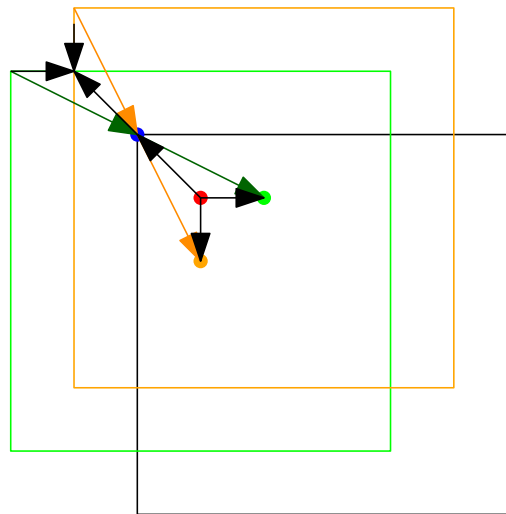


•  $U_0^P = \{u_m\}$

•  $U_1^P$

•  $\overline{U^P}$

(b) Průměr posunutých bodů dvou obrazů



•  $u_m$

•  $u_1^P$

•  $u_2^P$

•  $\overline{u^P}$

$\rightarrow T_1$

$\rightarrow T_2$

$\rightarrow$  nová  $T_i$

(c) Sesazení tří obrazů na průměr

Obrázek 9.2: Znázornění posuvů a sesazení na průměr

## 9.4. Poznámka o úplné korelaci

Ke hledání posuvu obrazových matic  $s$ ,  $r$  můžeme místo metody fázové korelace použít „obyčejnou“ diskretní cyklickou korelaci (4.9).

$$(\Delta x, \Delta y) = \operatorname{argmax}(s \times r) \quad (9.8)$$

Nyní spočítáme výpočetní náročnost obou těchto postupů. Příklad si nyní myšlenkově představíme v jedné dimenzi. Při fázové korelaci, viz (9.1), nám stačí spočítat třikrát Fourierovu transformaci. V případě, že počítáme pomocí rychlé Fourierovy transformace, máme náročnost  $\mathcal{O}(3N \log_2 N) = \mathcal{O}(N \log_2 N)$ . V případě diskretní cyklické korelace máme pro výpočet jednoho prvku sumu přes  $N$  prvků. Celkem tedy  $\mathcal{O}(N^2)$ . To znamená, že fázová korelace by byla podstatně rychlejší. Pro naši potřebu však algoritmus můžeme urychlit. Předpokládáme-li posuv o maximálně  $k$  pixelů na obě strany v obou rozměrech, nemusíme počítat všechny prvky, které bychom korelací dostali. Stačilo by jich  $(2k+1)^2$ , (tj. počet pixelů ve čtverci, kde je 0 uprostřed a můžeme se posunout o  $k$  na všechny strany). Výpočetní náročnost by byla  $\mathcal{O}(N(2k+1)^2)$ . V případě volby  $k=3$  dostaneme  $\mathcal{O}(49N) = \mathcal{O}(N)$ . I když konstanta 49 už bude hrát roli. V případě fázové korelace pro okolí uzlu velikosti  $N = 16 \times 16 = 256$  dostaneme  $\mathcal{O}(3N \log_2 N) = \mathcal{O}(24N) = \mathcal{O}(N)$ . Vidíme tedy, že výpočetní náročnost obou postupů je srovnatelná. Výpočet pomocí fázové korelace je nepatrně rychlejší (zanedbali jsme menší konstantu), ale není tomu tak pro velká okolí. Fázová korelace nám ale může dát posuv i o více pixelů. To může být v některých případech užitečné, ale v jiných nežádoucí.

## 9.5. Rozptyl hodnot a histogram

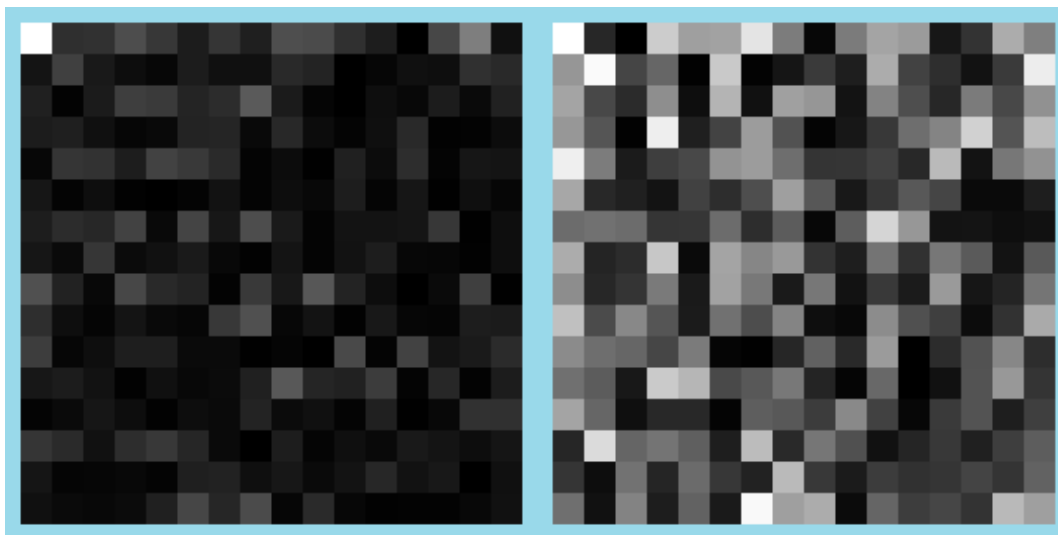
**Histogram** ukazuje četnosti výskytu hodnot funkce, která je definovaná na diskretním definičním oboru, v určitém intervalu. Při ekvidistantním dělení ho můžeme definovat následujícím způsobem.

**Definice 9.2.**

$$\begin{aligned} \mathbf{h} &= (h_1, \dots, h_n), & x &= x_0, x_1, \dots, x_L \\ h_i &= \sum_{f(x) \in I_i} 1, & \text{kde } I_i &= \left( \frac{(i-1)L}{N}, \frac{iL}{N} \right) \end{aligned} \quad (9.9)$$

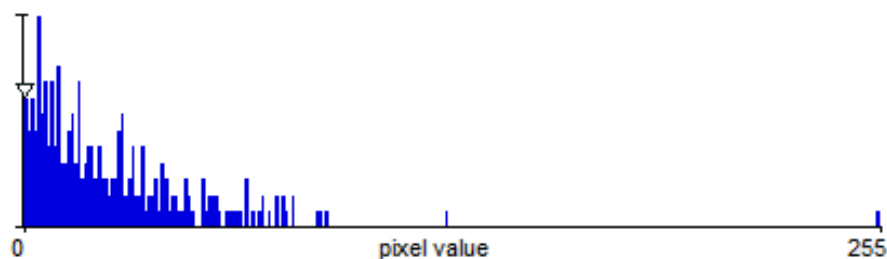
V našem případě normované křížové korelace si za funkci  $f(x)$  dosadíme obrazovou matici  $r = \mathcal{F}^{-1} \left\{ \frac{A\bar{B}}{|A\bar{B}|} \right\}$ . Pokud je maximum zřetelné, tak se hodnoty v  $r$  budou více blížit nulám, tedy četnosti v intervalech u nuly budou značně převyšovat četnosti v ostatních intervalech. Naopak pokud budou hodnoty  $r$  v intervalech zastoupeny rovnoměrněji, potom mezi nimi nemůžeme s jistotou říct, které maximum je náš posuv. To se může stát, například když na vstupních obrazech je téměř homogenní plocha (tj. nejsou zde žádné výrazné hrany, proto obraz může být posunutý kamkoliv) nebo pokud je obraz příliš zašumělý. V tom případě se histogram jeví vyrovnanější.

## 9.5. ROZPTYL HODNOT A HISTOGRAM

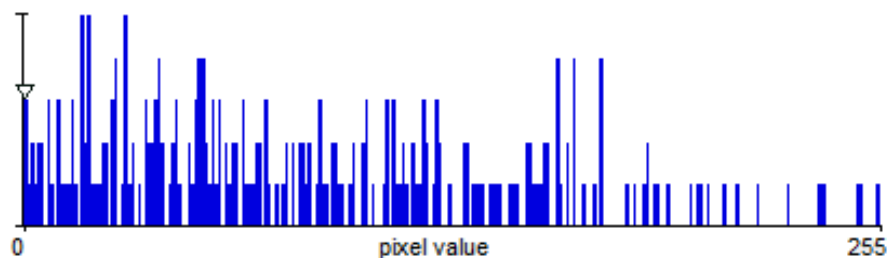


(a) Zřetelné maximum fázové korelace

(b) Nezřetelné maximum



(c) Histogram zřetelného maxima



(d) Histogram nezřetelného maxima

Obrázek 9.3: Zřetelné a nezřetelné maximum fázové korelace

Na obrázku 9.3a vidíme jasné maximum v bodě  $(0,0)$ , obrazy jsou tam tedy sesazené a nepotřebují další posuv. Odpovídající histogram 9.3c byl vytvořen až po převodu na bitmapu, kde pixely nabývají hodnot 0 až 255. (Původní hodnoty se tak rozdělili do 255 intervalů). Maximální hodnota je jenom jedna a hodně se liší od ostatních. Hodnot blízkých nule je hodně. Na obrázku 9.3b vidíme fázovou korelaci téměř homogenních ploch. Světlé body ukazují, že v případě posuvu do tohoto bodu by byly obrazy korelované. Proto je těžké určit, které lokální maximum odpovídá námi požadovanému posuvu.

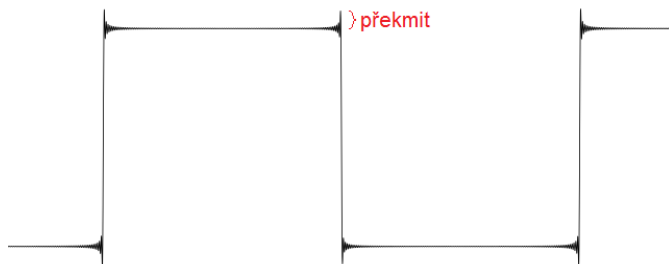
**Odchylka od nuly.** Lepším způsobem, než analyzovat histogram, je porovnávat odchylky hodnot  $r$  od nuly. Pro hodnoty  $r$  tedy předpokládáme střední hodnotu  $Er = 0$ . (V ideálním případě  $r = \delta(x - \Delta x, y - \Delta y)$  obsahuje samé nuly až na jeden bod, tímto zjistíme, jak moc se náš případ liší od ideálního). Rozptyl  $\sigma^2 = Dr = E(r - Er)^2 = Er^2 =$



$\frac{1}{N} \sum_{i=1}^N r_i^2$ , kde  $N$  je počet prvků v obrazové matici  $r$ . Odmocněním můžeme spočítat směrodatnou odchylku  $\sigma$ , ale pro potřebu algoritmu stačí porovnat  $\sigma^2 < limit$ . Když této podmínce bod vyhovuje, tak můžeme maximum považovat za významné. Hodnotu limitu je třeba stanovit experimentálně, vhodná hodnota se může lišit pro různé vstupní obrazové matice. Pokud bod podmínce nevyhovuje, nezahrneme ho mezi uzlové body, ze kterých získáme výslednou transformaci.

## 9.6. Okna

Při vyjádření Fourierovou transformací můžeme na okrajích dostat nežádoucí „překmitý“. To je způsobeno tím, že Fourierova transformace je tvořena skládáním harmonických kmitů. Tomuto jevu se říká *Gibbsův fenomén* nebo také *Gibbsův jev*.



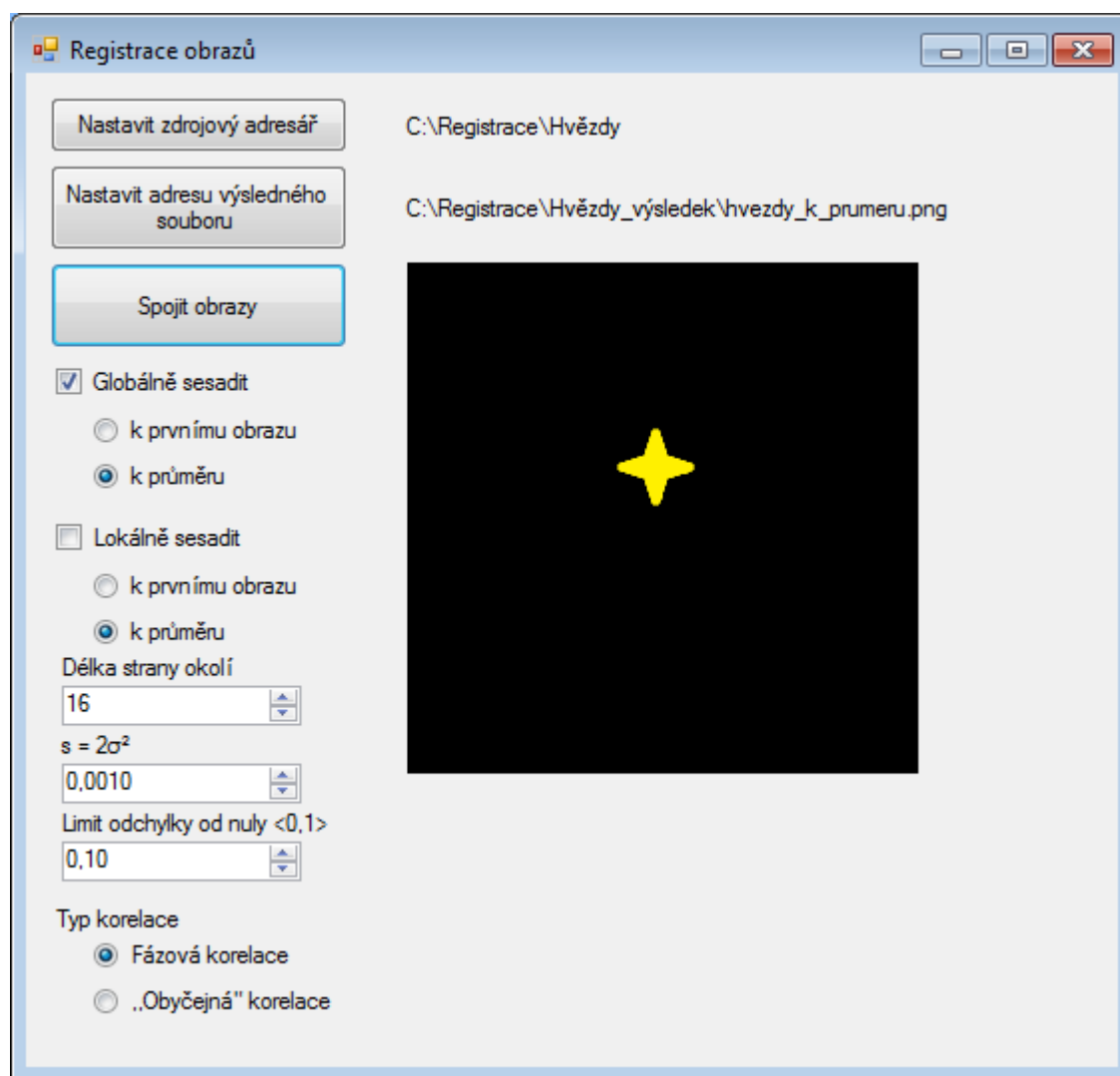
Obrázek 9.4: Gibbsův jev čtvercového pulsu

Na obrázku 9.4 vidíme součet několika prvních členů Fourierovy řady, která tvoří funkci čtvercového pulsu (tj. funkce, která periodicky nabývá pouze hodnot 1 a  $-1$ ). Strmá změna způsobuje Gibbsův jev. Při fázové korelaci může překmit vytvořit maximum, které není hledaným posuvem. Dvojměrná diskretní Fourierova transformace podle 5.7 předpokládá „periodicky se opakující“ matici. V případě diskretizované obrazové funkce nemůžeme předpokládat, že odpovídající obrazová matice je periodická. Nepředpokládáme, že data plynule navazují na data na protějším okraji matice. Proto je vhodné na ni aplikovat tzv. okno. Okno je funkce, pomocí které alespoň částečně nebo úplně potlačíme Gibbsův jev. Aplikování okna způsobí, že body uprostřed matice budou mít větší váhu, která se směrem k okraji plynule snižuje, to si můžeme představit podle obrázku 9.1b. Jako okno můžeme použít různé funkce, například výše uvedenou gaussovskou funkci nebo Hammingovo okno. Více o oknech nalezneme v [9].

# 10. Popis programu

Program je naprogramován v jazyce C# v prostředí Microsoft Visual Studio 2010. Pro jeho spuštění je nutné mít nainstalován .NET. Kompletní zdrojový kód je na přiloženém CD. Spustitelný soubor má adresu `\Registration\Registration\bin\Debug\Registration.exe`

## 10.1. Uživatelské rozhraní



Obrázek 10.1: Uživatelské rozhraní programu

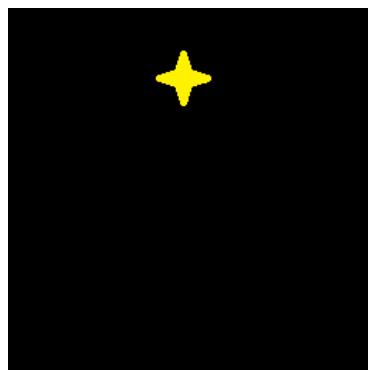
Obrázky nebo fotografie, které chceme sesadit, je třeba mít uloženy samostatně v jedné složce s příponou `.bmp`. Fourierova transformace se počítá algoritmem FFT, proto je nutné mít rozměry obrázku jako mocniny dvojky, nejlépe čtvercové. V případě že velikost není mocnina dvou, program obrázek automaticky zvětší, což však není vhodné, protože prázdné místo pouze nahradí černými pixely. To může mít vliv při počítání fázové korelace. V případě, že zadané obrazy nejsou stejně velké, tak program stejným způsobem menší obrazy zvětší. Obrázky mohou být uloženy barevně, program si je pro výpočet převede na černobílé a z nich udělá obrazové matice s komplexními hodnotami. Konečnou trans-

formaci pak provede z původních barevných. Transformované obrazy se uloží s vybraným názvem souboru, je-li jich více, program je očísluje. Když se obrazy sesazují globálně, tak se uloží i obraz zpětné transformace ze spektra, ve kterém je vidět maximum jako světlý bod. Globální sesazení se počítá vždy metodou fázové korelace, protože obvykle jsou třeba větší posuvy než jen o pár pixelů. Maximální posuv je v každém směru o polovinu velikosti obrazu. V případě lokálního sesazení máme možnost zvolit typ korelace. „Obyčejná korelace“ počítá s posuvem maximálně o tři pixely v obou směrech. Dále máme možnost volit parametr gaussovského jádra a limit významnosti maxima. Tyto hodnoty jsou v programu normované, tj. jsou vhodné pro všechny velikosti vstupních obrazů, proto je doporučuji neměnit. Dále máme možnost volit velikost okolí. To je v případě fázové korelace třeba volit jako mocninu dvou, v případě obyčejné korelace by mělo být větší než sedm, jinak mohou nastat problémy. Metoda zjišťování významnosti maxima je **!POMALÁ!**, proto testovat lokální sesazení doporučuji pouze pro jeden sesazovaný obraz a jeden referenční obraz o malých rozměrech. Výpočet pro větší obrazy může trvat i přes hodinu. V programu jsem bohužel neudělal zobrazování toho, kolik procent výpočtu je dokončeno. Algoritmus by se dal urychlit tak, že by se pro testování významnosti vybral pouze vzorek z okolních hodnot. Výsledek by ale nebyl tak přesný.

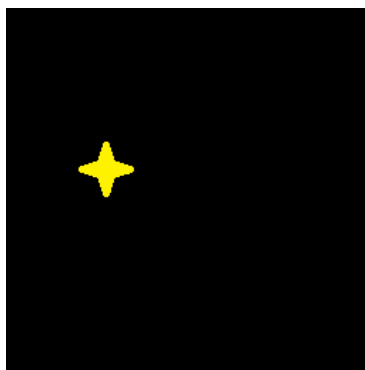
# 11. Výsledky

Výsledky použitých algoritmů demonstrujeme na několika příkladech. Zdrojové obrázky těchto příkladů jsou uloženy na přiloženém CD ve složce *Obrázky*.

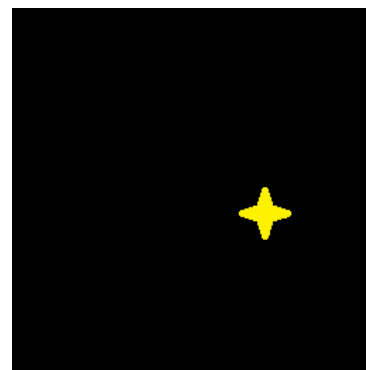
## 11.1. Globální sesazení více obrazů na průměr



(a) Referenční obrázek



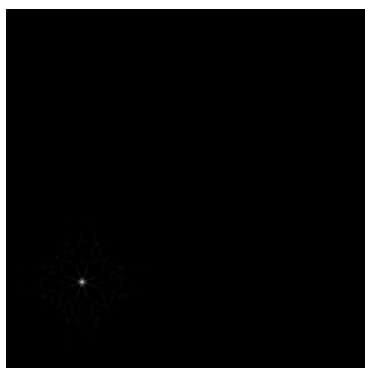
(b) První sesazovaný obrázek



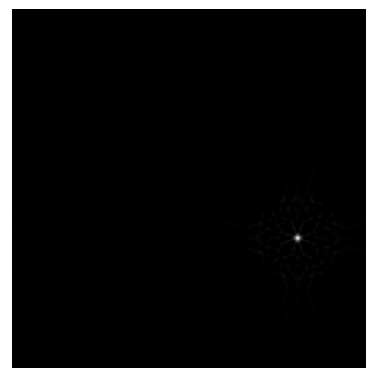
(c) Druhý sesazovaný obrázek



(d) Obrázek sesazený na průměr



(e) Matice fázové korelace prvního sesazovaného obrázku



(f) Matice fázové korelace druhého sesazovaného obrázku

Obrázek 11.1: Zřetelné a nezřetelné maximum fázové korelace

Výsledný obraz je v „těžišti trojúhelníka“. To můžeme vidět, když zdrojové obrazy i výsledek promítneme přes sebe.



Obrázek 11.2: Původní obrazy a výsledek v jednom

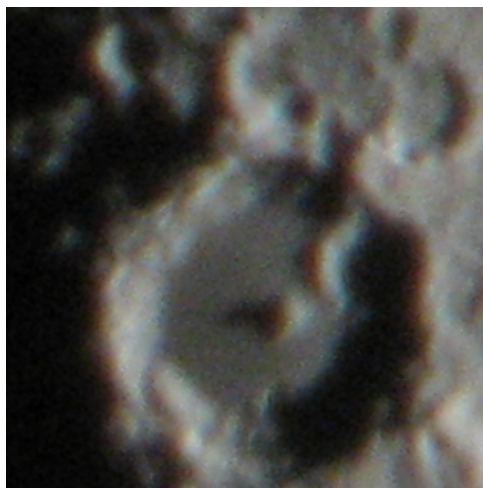
## 11.2. Lokální i globální sesazení zároveň, oba typy korelace

Pro porovnání výsledků obyčejné a fázové korelace jsem nastavil stejně parametry: velikost okolí 8pixelů,  $s = 0,001$ ,  $limit = 0,1$ , lokální i globální sesazení na průměr. Použil jsem dva vstupní obrazy 11.3a a 11.3b. Nevím, jak budou rozdíly v obrázcích viditelné v tištěné verzi. Na přiloženém CD je jejich elektronická verze. Transformované obrazy oběma způsoby se nepatrně liší. Rozdíly jsou viditelné, když se obrazy zobrazí postupně na jednom místě monitoru. Pokud by obě metody našly stejný posuv, měly by obrazy být totožné. To nejsou, takže někde musí docházet k drobným numerickým nepřesnostem. Algoritmus by se dal zlepšit větším počtem uzlových bodů (jejich okolí se klidně mohou překrývat) a také jejich lepším rozmístěním. Pro nalezení zřetelnějšího maxima by bylo vhodné umístit je na hrany objektů, ale ty by se musely detekovat.

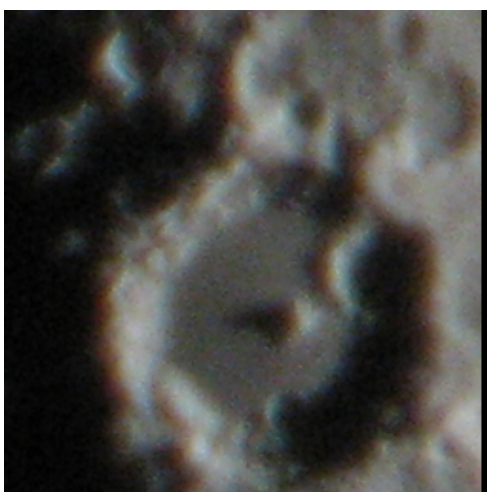
## 11.2. LOKÁLNÍ I GLOBÁLNÍ SESAZENÍ ZÁROVEŇ, OBA TYPY KORELACE



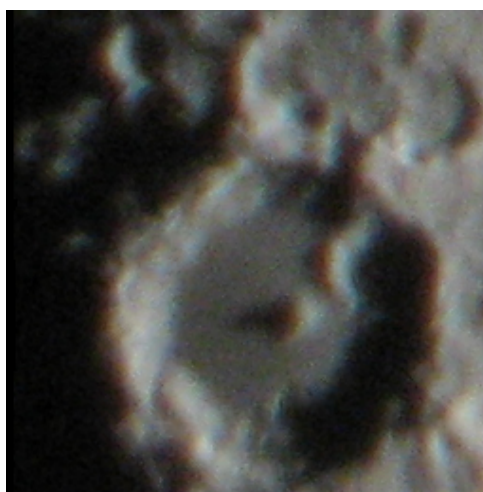
(a) První vstupní obrázek



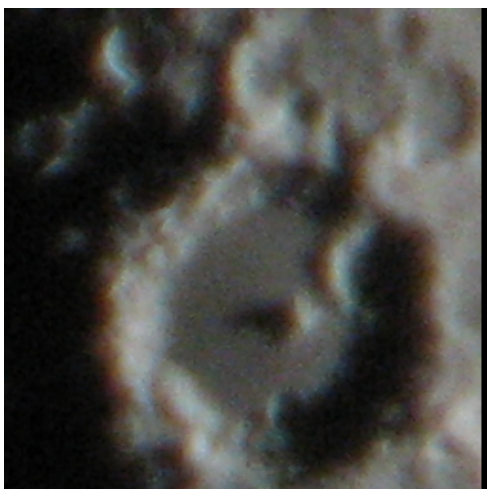
(b) Druhý vstupní obrázek



(c) První obrázek lokálně sesazen fázovou korelací



(d) Druhý obrázek lokálně sesazen fázovou korelací



(e) První obrázek lokálně sesazen obyčejnou korelací



(f) Druhý obrázek lokálně sesazen obyčejnou korelací

Obrázek 11.3: Obrazy měsíce

## 12. Závěr

Práce v úvodních kapitolách definuje lineární operátory. Ty mají kromě obrazové analýzy podstatný význam při zpracování signálů.

Stěžejní částí je odvození algoritmu fázové korelace. Z ní pak vychází rychlejší postup při hledání transformací.

Algoritmy uvedené v kapitole 9 se mi podařilo úspěšně implementovat v programu. Dosažené výsledky transformací jsou celkem příznivé. Předpokládal jsem, že algoritmus bude ještě rychlejší. S nabytými znalostmi bych výsledky nyní dokázal ještě zlepšit. Ovšem lepší výsledky se projeví delším výpočtem. Tyto výpočty by však bylo možné paralelizovat. Vypracováváním této práce jsem získal mnohem lepší představu o Fourierově transformaci, to se dá využít i v jiných vědeckých oblastech. Po programátorské stránce to pro mě byl také velký přínos.

# Literatura

- [1] FRANCŮ, Jan. *Funkcionální analýza I*. Vyd. 1. Brno: Akademické nakladatelství CERM, 2009, 149 s. ISBN 978-80-214-4015-9.
- [2] HLAVÁČ, Václav. *Fourierova transformace v 1D a 2D* [online]. [cit. 2015-05-28]. Dostupné z: <http://cmp.felk.cvut.cz/~hlavac/TeachPresCz/11DigZpr0br/12FourierTxCz.pdf>
- [3] KLÍMA, Miloš, Martin BERNAS, Jiří HOZMAN a Pavel DVOŘÁK. 1996. *Zpracování obrazové informace*. 1. vyd. Praha: Vydavatelství ČVUT, 177 s. ISBN 80-010-1436-3.
- [4] PRATT, William K. *Digital Image Processing (Third Edition) PIKS Inside*. 3rd ed. New York: Wiley-Interscience, 2001. ISBN 0471221325.
- [5] REDDY, B.S. a B.N. CHATTERJI. An FFT-based technique for translation, rotation, and scale-invariant image registration. *IEEE Transactions on Image Processing* [online]. 1996, vol. 5, issue 8, s. 1266-1271 [cit. 2015-05-28]. DOI: 10.1109/83.506761.
- [6] SARVAIYA, Jignesh, Suprava PATNAIK a Kajal KOTHARI. Image Registration Using Log Polar Transform and Phase Correlation to Recover Higher Scale. *Journal of Pattern Recognition Research* [online]. 2012, vol. 7, issue 1, s. 90-105 [cit. 2014-11-01]. DOI: 10.13176/11.355.
- [7] VESELÝ, Vítězslav. *Funkcionální analýza II*. Dostupné z: <http://www.econ.muni.cz/~vesely> [online]. [cit. 2015-01-16]
- [8] WILLIAMS, Lance. *Fourier Transform Theorems* [online]. [cit. 2015-05-28]. Dostupné z: <http://www.cs.unm.edu/~williams/cs530/theorems6.pdf>
- [9] Window function. 2001-. *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation [cit. 2015-05-28]. Dostupné z: [http://en.wikipedia.org/wiki/Window\\_function](http://en.wikipedia.org/wiki/Window_function)



# 13. Seznam použitých zkratek a symbolů

$\bar{f}$	konjugace
$Rf$	reflexe
$\tilde{f}$	involute
$f * g$	konvoluce
$f \times g$	korelace
$\mathcal{F}\{f\}$	Fourierova transformace
$\mathcal{F}^{-1}\{F\}$	zpětná Fourierova transformace
$a, b$	Fourierovsky netransformované obrazové matice
$A, B$	Fourierovsky transformované obrazové matice
$f, g$	Fourierovsky netransformované funkce nebo posloupnosti
$F, G$	Fourierovsky transformované funkce
$\mathbb{R}, \mathbb{C}$	množiny reálných a komplexních čísel

## 14. Seznam příloh

Na přiloženém CD se nachází program *Registration* včetně kompletního zdrojového kódu. Popis programu viz kapitola 10. Vedle programu jsou i přiloženy obrázky z příkladů kapitoly 11. Vše je uloženo v archivu *program.zip*. Součástí programu je i jedna převzatá knihovna, pomocí které se počítá rychlá Fourierova transformace. Je jí *AForge.NET framework*, z něj konkrétně je použita použita knihovna *AForge Image Processing Library*.

AForge.NET framework. 2009. KIRILLOV, Andrew. [online]. [cit. 2014-11-01]. Dostupné z: <http://www.aforgenet.com/framework/>